REMARKS

Claims 1-20 are pending in the application. Claims 1-20 have been rejected.

Claims 1-20 are rejected under 35 U.S.C. § 101. Independent claims 1, 9 and 15 have been amended to overcome this rejection. Claim 15 has additionally been amended to recite a physical computer readable storage medium as suggested by the Examiner.

Claims 5, 13 and 19 stand rejected under 35 U.S.C. §112, second paragraph. Claims 5, 19 have been amended to address this rejection. It is believed that claim 13 provides antecedent basis for the limitation of a property key name.

Claims 1-20 stand rejected under 35 U.S.C. § 102(e) as being anticipated by Warshavsky et al., U.S. Patent No. 6,732,095 B1 (Warshavsky). Claims 8, 14 and 20 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Warshavsky in view of Bradley et al, U.S. Patent No. 6,584,507 B1 (Bradley). These rejections are respectfully traversed.

The present invention, as set forth by independent claim 1, relates to a team sharing environment comprising an integrated development environment for persisting resource properties during transitions of data between a user and a team repository wherein the integrated development environment includes a property file for storing property keys and their associated resource property values.

The present invention, as set forth by independent claim 9, relates to a team sharing environment comprising a method for persisting resource properties in an integrated development environment during transitions of data between a user and a team repository which includes the step of storing, in a property file, a list of property keys to be persisted and their associated resource property values.

The present invention, as set forth by independent claim 15, relates to a computer program product having a computer readable medium tangibly embodying computer executable code for directing an integrated development environment to persist resource properties in a team sharing environment during transitions of data between a user and a team repository, the

computer program product includes code for storing, in a property file, a list of property keys to be persisted and their associated resource property values.

Warshavsky discloses a method to convert data between a relational format and an XML document. The method of Warshavsky creates a set of XML Mapping Definition from metadata; selects relational data from a relational application database, and converts the relational data to the XML document using the set of XML Mapping Definition.

When discussing the element of a property file for storing property keys and their associated resource property values, the Examiner sets forth:

> With respect to independent claim 1, Warshavsky clearly teaches in a team sharing environment, an integrated development environment for persisting resource properties during transitions of data between a user and a team repository (see column 4, lines 38-56), the integrated development environment comprising: a property file for storing property keys and their associated resource property values (see column 5, lines 4-30 and Tables 1-3; Note the property keys are the items listed under property name in Table 1-3 and the property values is the data that resides in those fields.). (Office action dated January 11, 2007, page 5.)

The portion of Warshavsky to which the Examiner refers sets forth:

> XML Mapping Definition 114 consists of three entities: Object, Component, and Field. An Object identifies a specific group of tables and a single XML document to be mapped. The Object contains global information, such as the document's root XML element name. Each Object has a set of components where these components are organized in a hierarchy that can have only one root component.
>
> A Component defines a mapping between a relational table and XML elements. Two XML elements may be specified for the table: one for the individual records and an optional element to group records belonging to the table. A Component contains zero or more fields. A Field defines the mapping between a column in the Component's table to either an XML element or an XML attribute. The fields within a component may map to a hierarchy of elements and attributes in the XML document.
>
> The XML Mapping Definition 114 may be automatically populated through use of the Metadata Wizard 115. In one embodiment, the Metadata Wizard 115 is an XML Metadata Wizard and the XML portion of the mapping is fixed by the external metadata, but the default relational portion can be defined by the XML Metadata Wizard. The XML Metadata Wizard may either define a simple mapping where each element of the XML document 104 is associated with a table or it may collapse portions of the XML hierarchy to minimize the number of tables needed to hold the data (Warshavsky Col. 5, lines 4 – 30).

However, nowhere in this portion of Warshavsky, nor anywhere else in Warshavsky is there any disclosure or suggestion of an integrated development environment, much less an integrated development environment which comprises a property file for storing property keys and their associated resource property values as disclosed and claimed.

More specifically, Warshavsky and Bradley, taken alone or in combination, do not teach or suggest a team sharing environment comprising *an integrated development environment* for persisting resource properties during transitions of data between a user and a team repository much less such an integrated development environment where *the integrated development environment includes a property file for storing property keys and their associated resource property values*, all as required by claim 1. Accordingly, claim 1 is allowable over Warshavsky and Bradley. Claims 2 - 8 depend from claim 1 and are allowable for at least this reason.

Warshavsky and Bradley, taken alone or in combination, do not teach or suggest a team sharing environment comprising a method for persisting resource properties in *an integrated development environment during transitions of data between a user and a team repository*, much less such a method which includes the step of *storing, in a property file, a list of property keys to be persisted and their associated resource property values*, all as required by claim 9. Accordingly, claim 9 is allowable over Warshavsky and Bradley. Claims 10-14 depend from claim 9 and are allowable for at least this reason.

Warshavsky and Bradley, taken alone or in combination, do not teach or suggest a computer program product having a computer readable medium tangibly embodying computer executable code for directing *an integrated development environment to persist resource properties in a team sharing environment during transitions of data between a user and a team repository*, much less such a the computer program product which includes *code for storing, in a property file, a list of property keys to be persisted and their associated resource property values*, all as required by claim 15. Accordingly, claim 15 is allowable over Warshavsky and Bradley. Claims 16 - 20 depend from claim 15 and are allowable for at least this reason.

Bradley discloses linking external information to a network management system. A network management system is installed for and executes in association with a managed network. An external application program is identified by defining and storing in a connection

file information that describes: the name and location of the program; a position in a menu control tree into which folders and items, which identify functions and options of the external application program, should be displayed and accessed; security roles associated with each folder and item; and other meta-information about the application program and its maker. The information may be stored in a markup format in a connection file. The network management system reads the connection file and integrates the information into its registry and other locations that determine how the network management system operates.

The cites to a number of portions of Bradley including the following portion

> A connection between the network management system and a 3rd-party application is established by creating a connection file that stores information about the connection. To create a connection file, a user selects the ADMIN option 204e, Management Connection sub-option 208a, and the "Create" option 216a within control tree 203 (Bradley, Col. 10, lines 25 30).

However, none of these portions of Bradley disclose or suggest where the environment further comprises an extension point for providing an application program interface to third party plug-ins for creating a property file for the third party plug-in as set forth in Claims 8, 14 and 20.

CONCLUSION

In view of the amendments and remarks set forth herein, the application is believed to be in condition for allowance and a notice to that effect is solicited. Nonetheless, should any issues remain that might be subject to resolution through a telephonic interview, the examiner is requested to telephone the undersigned.

The Commissioner is authorized to deduct any additional fees which may be necessary and to credit any overpayment to Deposit Account No. 090461.

I hereby certify that this correspondence is being electronically submitted to the COMMISSIONER FOR PATENTS via EFS on April 10, 2007.

/Stephen A. Terrile/

Attorney for Applicant(s)

Respectfully submitted,

/Stephen A. Terrile/

Stephen A. Terrile
Attorney for Applicant(s)
Reg. No. 32,946